PRIS at TAC2011 KBP Track

Yan Li, Xiaoning Li, Hanying Huang, Yang Song, Cheng Chang, Liaoming Zhou
Jing Xiao, Dian Yu, Weiran Xu, Guang Chen, Jun Guo
School of Information and Communication Engineering
Beijing University of Posts and Telecommunications
buptly@yahoo.com.cn

Abstract

Our method to Knowledge Base Population at TAC2011 is described in this paper. Rule-based method and machine learning method are combined in the Slot Filling task. And in the Entity Linking task, query expansion method, rule-based method and entity disambiguation method are mainly used.

1 Introduction

The main goal of the Knowledge Base Population (KBP) track at TAC 2011 is to promote research in and to evaluate the ability of automated systems to discover information about named entities and to incorporate this information in a knowledge source. Actually, it is not new for us as we have taken part in the KBP track for two years. We participated in both slot filling and entity linking tasks this year just as before.

The Slot Filling task involves learning a pre-defined set of relationships and attributes for target entities based on the documents in the test collection. Similar with our last year's work, rule-based method and machine learning method are both used in our system. But this year we set a more elaborate rule template and the Conditional Random Field algorithm and the Maximum Entropy algorithm are also involved.

The Entity Linking task is to determine for each query, which knowledge base entity is referred to, or if the entity is not present in the reference KB. And the main difficulties of this task are alias detection (that multiple queries may refer to the same entity using different name variants or different doc ids) and entity disambiguation (that the same query name may refer to multiple entities).

In TAC2010-KBP track, we consider Entity Linking task as a retrieval task. In order to resolve the two difficulties mentioned above effectively, we designed some rules for helping make better decisions. In all, three methods are employed in the entity-linking task, two basic retrieval models and another method we mainly focused on. One of the basic retrieval models corresponds to an optional task of entity linking without Wikipedia pages. The rule-based method is remained in our system this year. In addition, a new query expansion method and entity disambiguation method are also applied in this year's work. It is required to cluster together queries referring to the same non-KB (NIL) entities this year. And we simply use some rules to accomplish it.

The remaining of this paper is organized as follows. Section 2 and 3 describe systems about Slot Filling task and Entity Linking task respectively. Section 4 presents our evaluation results of the tasks.

2 Slot Filling Task

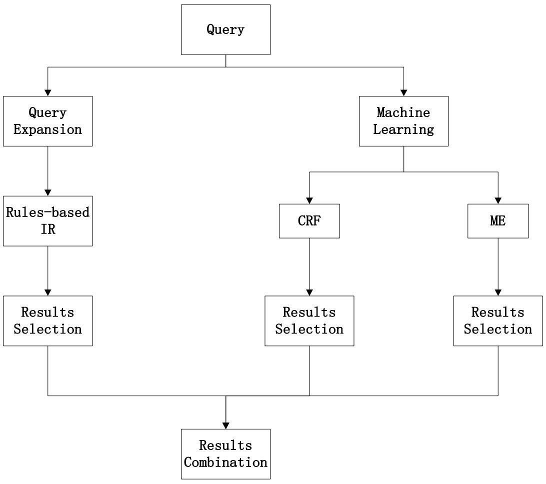


Figure 1: framework of Slot Filling system

Fig.1 shows the framework of our Slot Filling system, which consists of two branches: rule-based method and machine learning method. And finally the results of two parts are combined.

2.1 Rule-based IE

It is needed to expand the query first using the strategy as follows:

(1) For the entities of PER type, if the query is not the person's full name, we search for his/her full name in the supporting document.

(2) For the entities of ORG type, if the query is an abbreviation or full name, we search for its full name or abbreviation respectively in the supporting document.

Then we index the relative documents of the queries using these expansions as well as the queries themselves.

For rule-based information extraction, a set of rule patterns are designed to help filling the slots. Each rule pattern is a regular expression, which is mainly composed of four parts as shown in Tab.1: Target Type (type of the target entity), Slot (the slot name), Domain Type (type of the relation answer), and Keywords (typical words related to the slot). The Target Type and Domain Type are recognized first, and then pre-defined rule patterns are used to extract slots sentence by sentence.

2.2 CRF-based and ME-based Machine Learning

Fields	Descriptions
Target Type	PER (person) and ORG (organization)
Slot	The slots to be filled for the task. There are 26 slots for person and 16 slots for organization.
Domain Type	All the slot values are categorized into 12 domain types, which are shown in Tab.2. PER, ORG, and LOC are recognized by Stanford NER. DATE, URL and NUM (number) are recognized by regular expressions. Domain types for ORIGIN, DEATH, SCHOOL, TITLE, RELIGION and CHARGE are mainly from lists of candidates which come from the training data in KB.
Keywords	Each slot has one or more keywords, which are important for relation extraction.

Table 1: composition of a rule pattern

PER		ORG	
Domain	Slots	Domain	Slots
PER	per:alternate_names; per:spouse; per:children; per:parents; per:siblings; per:other_family	PER	org:alternate_names; org:members; org:shareholders; org:founded_by; org:top_members/emplyees
ORG	per:member_of; per:employee_of	ORG	org:parents; org:members; org:member_of; org:shareholders; org:subsidiaries
LOC	per:country_of_birth; per:stateorprovince_of_birth; per:city_of_birth; per:country_of_death; per:stateorprovince_of_death; per:city_of_death; per:countries_of_residence; per:stateorprovinces_of_residence; per:cities_of_residence; per:member_of; per:employee_of	LOC	org:member_of; org:city_of_headquarters; org:country_of_headquarters; org:stateorprovince_of_headquarters
DATE	per:date_of_birth; per:date_of_death	DATE	org:founded; org:dissolved
NUM	per:age	NUM	org:number_of_employees/members
ORIGIN	per:origin	URL	org:website
DEATH	per:cause_of_death	RELIGION	org:political/religious_affiliation
SCHOOL	per:schools_attended		
TITLE	per:title		
RELIGION	per:religion		
CHARGE	per:charges		

Table 2: Domain Type and Slots

The algorithm of Conditional Random Fields (CRFs) and Maximum Entropy (ME) are both applied for our machine learning information extraction method.

CRFs are a framework for building probabilistic models to segment and label sequence data. We can also think of a CRF as a finite state model with un-normalized transition probabilities. CRFs assign a well-defined probability distribution over possible labeling, trained by maximum likelihood or MAP estimation.

In Bayesian probability, the principle of ME is a postulate which states that, subject to known constraints (called testable information), the probability distribution which best represents the current state of knowledge is the one with largest entropy. Let some testable information about a probability distribution function be given. Consider the set of all trial probability distributions that encode this information. Then, the probability distribution that maximizes the information entropy is the true probability distribution with respect to the testable information prescribed. ME means that when you know nothing about the event, then it chooses a model to make its distribution as even as possible. Intuitively speaking, the toolset fits all known facts and keeps the unknown events unknown. In other words, given some fact sets, the toolset chooses a model consistent with the existing facts and makes the distribution of unknown events as even as possible.

For our Slot Filling task, we use the Maxent++ Toolset and the CRF++ Toolbox respectively. To improve the performance of the classifiers, the training data were divided into two parts according to the entity type (PER or ORG). And each of two parts was further divided into two and four smaller parts respectively according to whether the second token of the token pair is a named entity or not. The feature template we use for ME is the same, which includes:

(1) PF: A token pair, the first token is the target entity and the second is the relation token.

(2) SF: sequence feature, the sequence between the two tokens.

(3) EOF: entity location features, the position of the target named entity in the sentence.

(4) AF: appearance feature, if the token pair appears in the same sub-sentence, the feature is 1.

(5) NF: number feature, the number of words between the two tokens.

(6) EF: entity feature: if there is another named entity between the token pair, the feature is set to 1.

(7) TF: type feature, the entity type of the token pair, such as PER or ORG.

In the end, for every slot, the answer may not be single. So we have to clear up these answers by the confidence score. For the single-answer slot, we select the one with the highest score; while for the list-answer slot, we keep the answers with top three scores.

3 Entity Linking Task

The framework of Entity Linking system is shown in Fig.2. The EL-RMB system is carried on our last year's work. The WAF-based query expansion and the disambiguation method are introduced following.

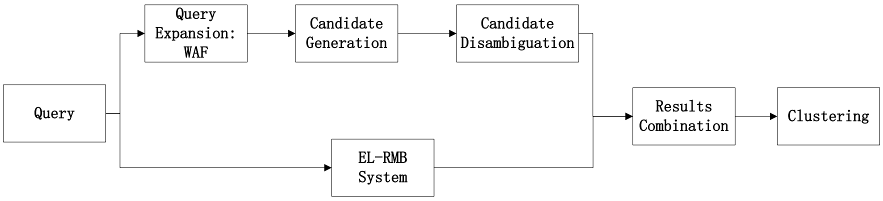


Figure 2: framework of Entity Linking system

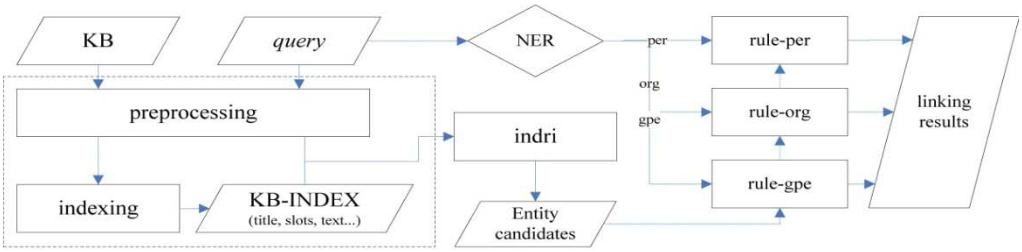


Figure 3: framework of the EL-RMB system

3.1 The EL-RMB System

In this year, we generally follow our previous work on entity linking 2010. We apply the same framework as shown in Fig.3.

On the whole, EL-RMB system consists of three primary parts: data preprocessing, candidate entities retrieval and linking decision. They are performed sequentially in the system. KB and source corpus are processed first, and then candidate entities are refined by Indri which is a language model-based information retrieval tool. Finally, target entity is determined based on a couple of rules.

We apply the same set of techniques as those in our entity linking 2010 work for the most part of EL-RMB system. However, some strategies are changed as well as some methods are drawn into the EL-RMB system to improve performance. As for finding the acronyms, we limit the length of query with all capital letters. That means query will not be regarded as acronym if its length beyond our threshold.

This year we find full names of some abbreviations and expand the original query with them. After analyzing a large number of KB nodes, we find that if a query in index only returns one KB node, for the most part the KB is actually we find. We bring this rule into this year’s task. In addition, in the last year’s result we find that the score of NIL type in person category went beyond the golden standard. For the sake of better result, we modify the rule for person category.

3.2 WAF-based Query Expansion

Firstly, we give a brief introduction about Word Activation Forces Map Word Networks (WAF).

Words associate with each other in a manner of intricate clusters. One believes that the activation strength from one word to another forges and accounts for the latent structures of the word networks. This implies that mapping the word networks from brains to computers, which is necessary for various purposes, may be achieved through modeling the activation strengths. Specifically, given the frequencies fi and fj and co-occurrence frequency fij of a pair of words i and j in the corpus used to simulate the language experience of the target human, we predict the strength of the activation that word i exerts on word j through:

$$\mathrm{waf} = \frac{\left(\frac{\mathrm{f_{ij}}}{\mathrm{f_i}}\right)\left(\frac{\mathrm{f_{ij}}}{\mathrm{f_j}}\right)}{\mathrm{d_{ij}}^2} \qquad (1)$$

where $\mathrm{d_{ij}}$ is the average distance by which word i precedes word j in their co-occurrences. Seeing the ratios of $\mathrm{f_{ij}}$ to $\mathrm{f_i}$ and $\mathrm{f_{ij}}$ to $\mathrm{f_j}$ as masses, we identify that the statistic is defined in the same form of the universal gravitation. Therefore we name it as word activation force from i to j, shortly $\mathrm{waf_{ij}}$. Readily, given a vocabulary, the wafs of every pair of the words constitute a squared but asymmetrical matrix WAF = {wafij}, i.e. a directed word network, where nonzero elements in the ith row give the out-links of the ith node (from word i to others), while nonzero elements in the ith column the in-links of it (from others to word i). To identify word clusters based on the distinctive directed word network WAF, we introduce a word affinity measure $\mathrm{A^{waf}}$ from a unique perspective that deviates from the currently popular ones of semantic space models (perspectives of vector space). $\mathrm{A^{waf}}$ is defined as the geometric average of the mean overlap rates of the in-links and out-links of the inquired two words. For a pair of words i and j in the directed word network WAF, we define their affinity as:

$$A_{ij}^{waf}[\frac{1}{|K_{ij}|}\sum_{k\in K_{ij}} OR(waf_{ki}, waf_{kj})\frac{1}{|L_{ij}|}\sum_{I\in L_{ij}} OR(waf_{il}, waf_{jl})]^{1/2} \quad (2)$$

where $K_{ij}=\{k|waf_{ki}>0$ or $waf_{kj}>0\}$, $L_{ij}=\{l|waf_{il}>0$ or $waf_{jl}>0\}$, and $OR(x,y)=min(x,y)/max(x,y)$. Readily, K_{ij} and L_{ij} are the sets of the labels of the words connected by the in-links and the out-links of word i or word j, respectively. And OR(x,y) is an overlap rate function of x and y. using this measure we can acquire an undirected word network whose links represent word affinities from the directed one WAF.

For our entity linking task, we use the source data and supporting documents to build a relationship between words based on the theory above. After computing the final A^{waf} matrix, for a term of the initial query, we regard the word that has the largest A^{waf} with it as the expanded term.

3.3 Candidate Generation and Disambiguation

Firstly, we search the index of KB nodes with the initial queries and their corresponding terms expanded in section 3.2. Then we select top 10 KB nodes for each query and get a set of candidate KB nodes. If the candidate set is empty, which means that we cannot find any candidate KB node, we simply return NIL as the answer. If the candidate set only contains one item, we decide it as the final answer. When there are multiple items in the set, we should disambiguate the candidates and find the most probable one as the answer. Meanwhile we set a threshold to distinguish NIL and non-NIL if possible.

We implement the basic vector space model (VSM) approaches to achieve the goal. The intuition behind the VSM is that the more similar (based on word co-occurrence information) between the KB text with the context of the query, the more likely the KB node refers to the query. We use the Cosine Similarity to find out how similar they are.

The Cosine Similarity approach can be described as follows: If the KB text and the query context are denoted T1 (a1, a2, a3, … , am) and T2 (b1, b2, b3, …, bm) respectively, where m is the word space of the VSM and the elements of the vector stand for term frequency, then the Cosine Similarity is computed as:

$$sim = \frac{\sum_{i=1}^{m} a_i \times b_i}{\sqrt{\sum_{l=1}^{m} a_l^2} \times \sqrt{\sum_{k=1}^{m} a_k^2}} \quad (3)$$

Now that we can get the similarity between each KB texts in the candidate set and the query context, if the value of the most similar one is bigger than the threshold, we decide it as the answer; otherwise we return NIL.

4 Evaluation Results

4.1 Evaluation Results of Entity Linking

Three runs were submitted for the Entity Linking task this year, and Tab.3 shows the evaluation results.

	B3-precision	**B3-recall**	**B3-F1**
pris1	0.426	0.430	0.428
pris2	0.436	0.445	0.440
pris3	**0.481**	**0.502**	**0.491**

Table 3: Entity Linking Task Evaluation Results

<table>
<tr><td colspan="2"># Retrieved: 1513
Wrong: 1288
Redundant: 18
Inexact: 51
Correct: 156</td></tr>
<tr><td><b>Precision</b></td><td>0.10310641 (156/1513)</td></tr>
<tr><td><b>Recall</b></td><td>0.16507937 (156/945)</td></tr>
<tr><td><b>F1</b></td><td>0.12693246</td></tr>
</table>

Table 4: Slot Filling Task Evaluation Results

4.2 Evaluation Results of Slot Filling

Three runs were submitted for the Slot Filling task this year, and Tab.4 shows the evaluation results.

PRIS at TAC2011 Guided Summarization Track

Jiayue Zhang, Cong Yao, Xiaojun Ding, Zhenpeng Li
Weiran Xu, Guang Chen, Jun Guo
School of Information and Communication Engineering
Beijing University of Posts and Telecommunications
jyz0706@gmail.com

Abstract

Our method to accomplish the Guided Summarization Track at TAC2011 is described in this paper. To produce summary for both set A and B, a topic based sentence clustering algorithm is applied first, and sentence ranking algorithm afterwards, finally the sentence ranked top in each cluster is extracted to form the result. To detect the novelty of Set B, a new similarity measuring method is introduced.

1 Introduction

The guided summarization task is to write a 100-word summary of a set of 10 newswire articles (set A) for a given topic, where the topic falls into a predefined category. Participants are given a list of important aspects for each category, and a summary must cover all these aspects. Additionally, an "update" component of the guided summarization task is to write a 100-word "update" summary of a subsequent 10 newswire articles (set B) for the topic, under the assumption that the user has already read the earlier articles. It is the first year for our group to participant the guided summarization task, so we applied our own ideas in the classic framework.

For both set A and B, to improve readability, sentence is viewed as the basic unit of the articles, and a sentence extraction method is used. To cover the important aspects, the sentences under the same topic are clustered first using topic based k-means clustering algorithm first, and then the sentences in a cluster are ranked according to several aspects, such as sentence location information in the related article. Finally, the sentence ranked top of each cluster is extracted to form the final summary. For set B, a new similarity measuring method is introduced.

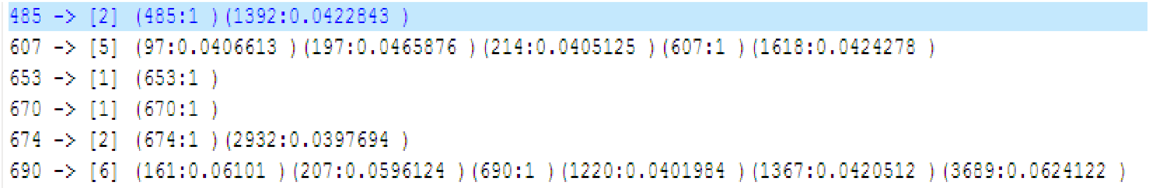
Compared with a whole article, a sentence is rather short. When using bag of words model, the sentence-word matrix can be very sparse which leads to poor performance of clustering or other algorithm. Our intuition: for different sentences under the same category, the words may differ greatly, but the subject or the topic may be the same, in other words, sentences may be represented by topics, and a topic including several words. With such representation, the sparse matrix problem is fixed. Topic model has been studied since 1999, such as PLSA, LDA, etc. Instead of those complicated model, in this paper, we introduce a new model to produce topic, Word Activation Forces Map Word Networks (WAF for short).

The remaining of this paper is organized as follows. Section 2 introduces the sentence clustering algorithm using WAF to produce words topic. Section 3 describes several sentence ranking algorithms. In section 4, the method of novelty detecting is presented.

2 Topic based clustering using WAF

In this part, we used a new method to make the similar words clustering which is named Word Activation Forces that is proposed by Professor Guo Jun in Beijing University of Posts and Telecommunications, here we use WAF for short. With WAF, the relevance between words and the affinity value between them can be measured. The larger the affinity value is, the stronger the conjunctions between words are.

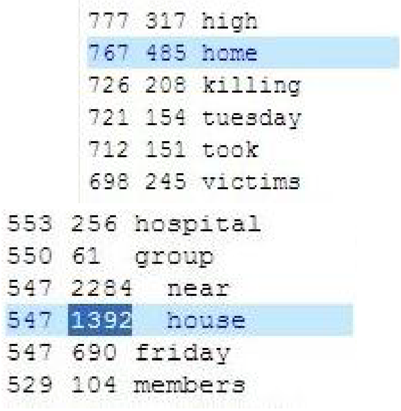
Firstly, given a topic, two thousand articles are retrieved about this topic in the AQUINT corpus as background. Then, a waf matrix is generated using WAF. It includes all of word activation forces between every two words in the two thousand articles. Secondly, the concerned words in the ten articles under the topic are abstracted after getting rid of the repetitive words and stop words, and checking their existence in the word list of the two thousand articles. Actually, most words are shared between the target documents and the two thousand background articles because they all belong to one topic. The new word list of topic creates an affinity value matrix using the waf-matrix which have generated. (Fig.1)



```
485 -> [2]  (485:1 )(1392:0.0422843 )
607 -> [5]  (97:0.0406613 )(197:0.0465876 )(214:0.0405125 )(607:1 )(1618:0.0424278 )
653 -> [1]  (653:1 )
670 -> [1]  (670:1 )
674 -> [2]  (674:1 )(2932:0.0397694 )
690 -> [6]  (161:0.06101 )(207:0.0596124 )(690:1 )(1220:0.0401984 )(1367:0.0420512 )(3689:0.0624122 )
```

Figure 1: Affinity value matrix

The number in first column represents the word of the word list, and the number in the square brackets represents that how many words have affinity values with the word. We have a lower limit of the affinity value between words which is decided by experience. From the affinity matrix, we can find the closest word of one word. The term information (Fig.2) will help to know the effect of WAF:

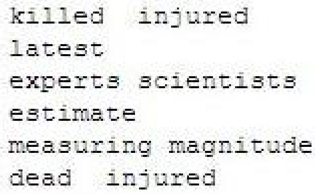


```
777 317 high
767 485 home
726 208 killing
721 154 tuesday
712 151 took
698 245 victims
553 256 hospital
550 61  group
547 2284  near
547 1392  house
547 690 friday
529 104 members
```

Figure 2: Term information

The first column represents the frequency of the word in the background articles, and the second column shows the id of word. From figure1 we can see the word whose id is 485 has a close word (except itself) whose id is 1392. It shows that 485 represent the word “home”, and 1392 represent the word “house” which is close to “home” from term information.

With the affinity value between the concerned words, the closest word of one word can be found, as in Fig.3.

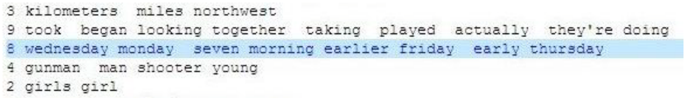


```
killed  injured
latest
experts scientists
estimate
measuring magnitude
dead  injured
```

Figure 3: The first-cluster term

In Fig.3, every row has two words at most, and some has only one word because of the lower limit in the process above. Due to WAF, every two words have an affinity value. Low affinity value means the two words are not close. Thus, words whose affinity value is lower than the limit should not be included in.

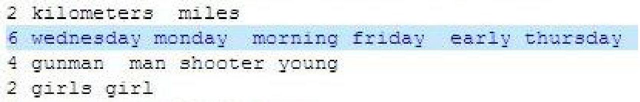
At last, a method that is similar to Hierarchical Clustering is used. From the first cluster we have got at most a closest word of one word, and the next step is to iterate every cluster to check whether two clusters have the same word. If there is a word in two different clusters, the two clusters will be merged into one cluster. The process will continue until there is no overlap in two different clusters. Figure4 shows the result of word-cluster for the ten articles in one topic.



```
3 kilometers  miles northwest
9 took  began looking together  taking  played  actually  they're doing
8 wednesday monday  seven morning earlier friday  early thursday
4 gunman  man shooter young
2 girls girl
```

Figure 4: Word-cluster result

In Fig.4, first column is the number of word in the cluster. From the result, we can see that the word in one cluster is close to each other mostly. For example, the cluster which highlight includes the days of a week those appear in the articles. But we also see “seven” that looks like having no relationship to other words. The reason is that “seven” has a lower affinity value with other word. When we raise the lower limit, the result changes (Fig.5):



```
2 kilometers  miles
6 wednesday monday  morning friday  early thursday
4 gunman  man shooter young
2 girls girl
```

Figure 5: The other word-cluster result

Now there no more “seven”, but the “earlier” is lost at the same time. The higher the lower limit is, the more precise the result becomes, but the more words we lost. So at times, the lower limit is reduced to save more useful words, and there isn’t a vastly bad effect on the accuracy of result. In this step, set A and B use the same clustering method.

Using the WAF method, we get clusters of key words. A cluster can be viewed as a concept, then all the key words together build up a concept space, each group of words representing a dimension of the space. Thus, sentences can be represented by vectors of concepts.Then, the difference between two sentences is calculated, with improved Minkowski Distance measure. If x_i and y_i represent the i_{th} dimension of the sentence x and sentence y, the difference between x and y is:

$$dist(x,y) = (\sum_{i=1}^{n} weight \times \left| x_i - y_i \right|^p)^{1/p} \quad (1)$$

where p is the total number of dimensions in the space.

The importance of each dimension in the space varies, so that some empirical coefficients (weight in equation 1) are introduced as the weights of different dimensions. Some person names, locations (entity attributes), together with verbs and nouns of high-frequency of, are usually part of important dimensions. When they are calculated, the coefficient introduced to enlarge the difference ranges from 1 to 3, according to test results.

Finally, for a specific document collection, a sentence is viewed as the minimum unit of the document, gathered together. The K-means approach is applied. In addition, the k value is determined, according to actual results.

The size of data given considered, one topic consists of about 10 documents, and one document contains 6 to 10 sentences. As a result, the number of sub-topics which stands for an independent meaning in the final summary is less than 12. The K value changes from 8 to 12, according to the later sorting algorithm.

3 Sentences Ranking Methods

In the previous step, the similarity between every two sentenceshas been calculated, and also, some clusters of sentences have been obtained. In each cluster, we would sort these sentences through their scores of similarity.

Three different methods were used to accomplish this task. In the first method, to calculate one sentence’s score, we just add each score of similarity between it and others. So the score we got is the final score of this sentence. After that, the only thing we need to do is to sort sentences through these final scores.

The second method we used is PageRank method (it is used by Google to identify the web pages importance).First, we constructed an n*n matrix in which each element is the similarity between two sentences. Second, the scores smaller than t (a threshold) would be set to 0, and others would be set to 1. (Here 0 denotes that there is no link, and 1 denotes a link) At last, PageRank method is used to calculate the final score.

In the third method, the two kinds of scores were combined by using a formula as follows,

$$S = i \cdot s1 + j \cdot s2 \quad (2)$$

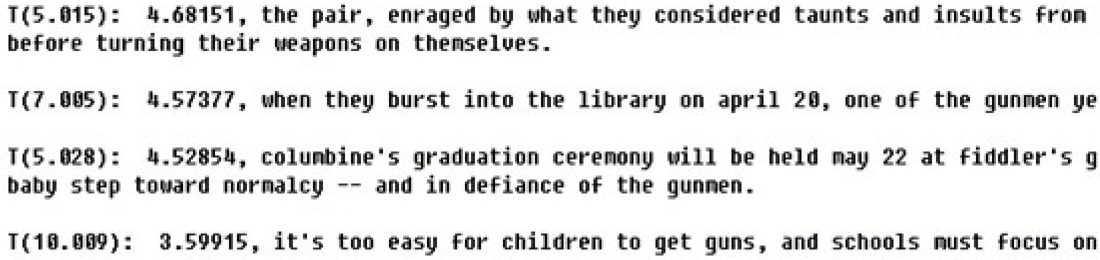
where i+j=1 and S1, S2 are the two scores we obtained from the above methods. The sentence location information was also taken into account, the final score is calculated as in equation 3.

$$Sf = S / \max(Si) + T / \max(Ti) \, . \quad (3)$$

Si refers to each S in the cluster and Ti refers to each sentence location in the text which it belongs to.

After ranking the sentences according to their scores, the top sentence is chosen to generate the summarization.

These figures show three different results of using different ways



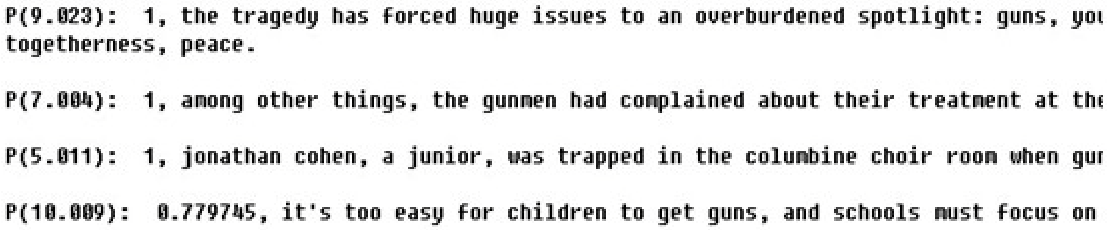
```
T(5.015):  4.68151, the pair, enraged by what they considered taunts and insults from
before turning their weapons on themselves.

T(7.005):  4.57377, when they burst into the library on april 20, one of the gunmen ye

T(5.028):  4.52854, columbine's graduation ceremony will be held may 22 at fiddler's g
baby step toward normalcy -- and in defiance of the gunmen.

T(10.009):  3.59915, it's too easy for children to get guns, and schools must focus on
```

Figure 6: Result of using the first method



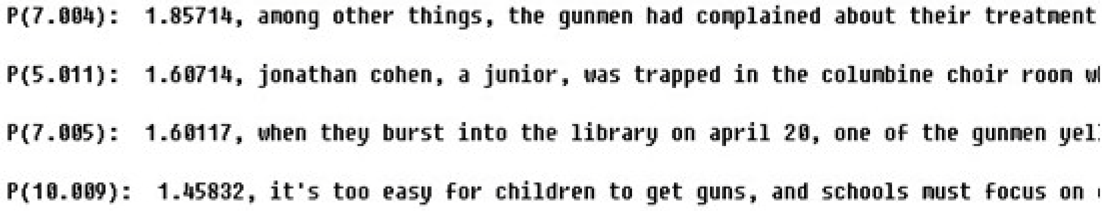
```
P(9.023):  1, the tragedy has forced huge issues to an overburdened spotlight: guns, you
togetherness, peace.

P(7.004):  1, among other things, the gunmen had complained about their treatment at the

P(5.011):  1, jonathan cohen, a junior, was trapped in the columbine choir room when gur

P(10.009):  0.779745, it's too easy for children to get guns, and schools must focus on
```

Figure 7: Result of using the second method



```
P(7.004):  1.85714, among other things, the gunmen had complained about their treatment

P(5.011):  1.60714, jonathan cohen, a junior, was trapped in the columbine choir room wl

P(7.005):  1.60117, when they burst into the library on april 20, one of the gunmen yel

P(10.009):  1.45832, it's too easy for children to get guns, and schools must focus on
```

Figure 8: Result of using the third method

4 Novelty detection

When it comes to the document set B, novelty is introduced as one of the characteristics of sentence selection. To capture that property we assume that every sentence is associated with an event, the topic that sentence represents. As a result, according to novelty, the second or third sentence about the same event is less interesting than the first. A new sentence is compared "its" event to that of all prior sentences, when it arrives. If it is different from all of other events, it will be considered as novel and given a high score. If e(s$_k$) represents the event discussed by sentence s$_k$, then the novelty of the k$_{th}$ sentence is:

$$P\left(novel\right)=\left[\prod_{i<k}(1-P(e(s_k)=e(s_i)))\right]^{\frac{1}{k-1}} \tag{4}$$

$$=\left[\prod_{i<k}(1-\left[\prod_{w\in s_k}\frac{tf\,(w,s_i)+0.01}{1.01\cdot\left|s_i\right|}\right]^{\frac{1}{|s_k|}})\right]^{\frac{1}{k-1}}$$

To generate a summary of set B, sentences in the set are gathered into several clusters first. Then, to obtain the novelty of every cluster, all sentences in the summary of set A are considered as prior sentences, and the center of the cluster (when K-means clustering approach is done, every cluster will get its center automatically) is the coming sentence, the novelty of the center sentence and the summary is obtained like Formula 4. Higher the score is, more likely the cluster can pick up its represent (some sentence in the cluster) to fill in the summary. Finally, sorting algorithm is carried on to generate the final summary.

References

X. Wang, H.Li, and J. Xiao. EUSUM: Extracting Easy-to-Understand English Summaries for Non-Native Readers. In proceedings of SIGIR, 2010 491-498.

B. Schiffman, A. Nenkova, K. Mckeown. Experiments in Multidocument Summarization. In proceedings of HLT, 2002.

L. Chong, M.Huang, X. Zhu. Tsinghua University at TAC 2009: Summarizing Multi-documents by Information Distance. In proceedings of TAC 2009.

D. R. Radev, H. Jing, M. Stys, D. Tam. Centroid-based summarization of multiple documents. Information Processing and Management 40 (2004) 919–938.

A. Haghighi, L. Vanderwende. Exploring Content Models for Multi-Document Summarization. Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the ACL, pages 362–370.

H. M. Wallach. Topic Modeling: Beyond Bag-of-Words. In Proceedings of the 23rd International Conference on Machine Learning, Pittsburgh, PA, 2006.

A. Gruber, M. Rosen-Zvi, Y. Weiss. Latent Topic Models for Hypertext.

Y. Kim. Document Clustering in a Learned Concept Space. Doctoral dissertation, l'Universite Pierre et Marie Curie, Paris, France, 2010.

Y. Zheng, T. Takenobu. The TITech Summarization System at TAC-2009. In proceedings of TAC 2009.

Guo, J., Guo, H. & Wang, Z. An Activation Force-based Affinity Measure for Analyzing Complex Networks. Sci. Rep. 1, 113; DOI:10.1038/srep00113 (2011).